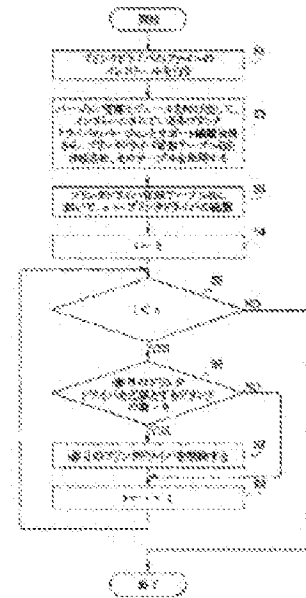


INFORMATION PROCESSOR AND METHOD, AND CONTROL PROGRAM**Publication number:** JP2004078744**Publication date:** 2004-03-11**Inventor:** EBATA MASAMICHI**Applicant:** CANON KK**Classification:****- international:** *B41J29/38; G06F3/12; G06F13/10; B41J29/38; G06F3/12; G06F13/10; (IPC1-7): G06F3/12; B41J29/38; G06F13/10***- European:****Application number:** JP20020240555 20020821**Priority number(s):** JP20020240555 20020821[Report a data error here](#)**Abstract of JP2004078744**

PROBLEM TO BE SOLVED: To automatically recognize an unnecessary printer driver, to provide know-how to delete the recognized driver, and to simplify the uninstalling operation of the printer driver.

SOLUTION: This information processor comprises, for example, an acquiring step for acquiring the version information of a device driver in a host computer 1000 and a printer driver management table B of information charts 11, 12 of peripheral devices capable of being controlled by the device driver, a recognizing step for recognizing the unnecessary device driver among the device drivers in which a plurality of versions can exist corresponding to the specific peripheral device, on the basis of the version information of the device driver acquired in the acquiring step, and a delete control step for controlling the delete of the device driver recognized as the unnecessary device driver by the recognizing means.

COPYRIGHT: (C)2004,JPO



Data supplied from the *esp@cenet* database - Worldwide

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-78744

(P2004-78744A)

(43) 公開日 平成16年3月11日(2004.3.11)

(51) Int. Cl.⁷

G06F 3/12
B41J 29/38
G06F 13/10

F I

G06F 3/12 C
B41J 29/38 Z
G06F 13/10 320A

テーマコード (参考)

2C061
5B014
5B021

審査請求 未請求 請求項の数 9 O L (全 14 頁)

(21) 出願番号	特願2002-240555 (P2002-240555)	(71) 出願人	000001007
(22) 出願日	平成14年8月21日 (2002. 8. 21)		キヤノン株式会社
			東京都大田区下丸子3丁目30番2号
		(74) 代理人	100090538
			弁理士 西山 恵三
		(74) 代理人	100096965
			弁理士 内尾 裕一
		(72) 発明者	江端 正道
			東京都大田区下丸子3丁目30番2号キヤ
			ノン株式会社内
		Fターム(参考)	2C061 AP01 HJ10 HQ01
			5B014 FA05 FA06
			5B021 AA01 CC06

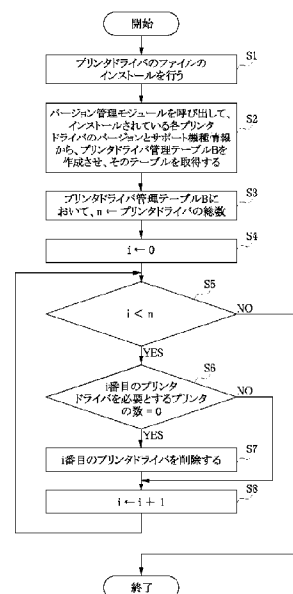
(54) 【発明の名称】 情報処理装置、情報処理方法、制御プログラム

(57) 【要約】

【課題】 不要となったプリンタドライバを自動的に認識し、認識したドライバを削除する仕組みを提供し、プリンタドライバのアンインストール作業を簡便化する。

【解決手段】 例えば、ホストコンピュータ1000内のデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報図11、図12のプリンタドライバ管理テーブルB)を取得する取得ステップと、前記取得ステップが取得したデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報をを用いて、特定の周辺装置に対応し、複数のバージョンが存在し得るデバイスドライバであって、情報処理装置の記憶装置に存在するデバイスドライバのうち不要となったデバイスドライバを認識する認識ステップと、前記認識手段によって不要と認識されたデバイスドライバの削除を制御する削除制御ステップを有することを特徴とする管理方法を提供する。

【選択図】 図6



【特許請求の範囲】**【請求項1】**

周辺装置に接続可能な情報処理装置にデバイスドライバを管理する管理方法であって、
情報処理装置内のデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報を取得する取得ステップと、
前記取得ステップにて取得したデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報をを用いて、特定の周辺装置に対応し、複数のバージョンが存在し得るデバイスドライバであって、情報処理装置の記憶装置に存在するデバイスドライバのうち不要となったデバイスドライバを認識する認識ステップと、
前記認識手段によって不要と認識されたデバイスドライバの削除を制御する削除制御ステップを有することを特徴とする管理方法。

10

【請求項2】

前記認識ステップは、周辺装置を特定する特定手順と、前記特定手順により特定された周辺装置を制御可能なデバイスドライバのうち、最もバージョンが新しいデバイスドライバを不要ではないと判断する判断手順とを備え、
前記特定手順により特定された周辺装置に対応するデバイスドライバに対して繰り返し前記判断手順における処理を行うことを特徴とする請求項1に記載の管理方法。

【請求項3】

前記認識ステップでは、周辺装置とデバイスドライバの情報を含むテーブルデータとして管理し、該テーブルデータより、不要となったデバイスドライバを求めることを特徴とする請求項2に記載の管理方法。

20

【請求項4】

新規にデバイスドライバを情報処理装置に追加した後に、前記取得ステップと、前記認識ステップと、前記削除ステップを、実行することを特徴とする請求項1乃至3のいずれかに記載の管理方法。

【請求項5】

周辺装置に接続可能な情報処理装置にデバイスドライバを管理可能な情報処理装置であって、
情報処理装置内のデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報を取得する取得手段と、
前記取得手段が取得したデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報をを用いて、特定の周辺装置に対応し、複数のバージョンが存在し得るデバイスドライバであって、情報処理装置の記憶装置に存在するデバイスドライバのうち不要となったデバイスドライバを認識する認識手段と、
前記認識手段によって不要と認識されたデバイスドライバの削除を制御する削除制御手段とを有することを特徴とする情報処理装置。

30

【請求項6】

前記認識手段は、特定の周辺装置を制御可能なデバイスドライバのうち最もバージョンが新しいデバイスドライバを不要ではないと認識することを特徴とする請求項5に記載の情報処理装置。

40

【請求項7】

前記認識ステップでは、周辺装置とデバイスドライバの情報を含むテーブルデータとして管理し、該テーブルデータより、不要となったデバイスドライバを求めることを特徴とする請求項6に記載の情報処理装置。

【請求項8】

新規にデバイスドライバを情報処理装置に追加した後に、前記取得手段と、前記認識手段と、前記削除手段とを、機能させることを特徴とする請求項6乃至7のいずれかに記載の情報処理装置。

【請求項9】

周辺装置に接続可能な情報処理装置にデバイスドライバを管理可能な情報処理装置におい

50

て実行される制御プログラムであって、
情報処理装置内のデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報を取得する取得ステップと、
前記取得手段が取得したデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報をを用いて、特定の周辺装置に対応し、複数のバージョンが存在し得るデバイスドライバであって、情報処理装置の記憶装置に存在するデバイスドライバのうち不要となったデバイスドライバを認識する認識ステップと、
前記認識ステップによって不要と認識されたデバイスドライバを削除する処理を制御する削除制御ステップとを情報処理装置において実行させることを特徴とする制御プログラム。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、周辺装置、周辺装置と接続可能な情報処理装置、及び、これらを含むシステムに関するものである。

【0002】

【従来の技術】

一般に、ホストコンピュータと、周辺装置の一例であるプリンタによって構成される印刷システムにおいて印刷を行なう場合、まず、ホストコンピュータ側でアプリケーションを実行して文字や図形などの印刷対象の描画データを作成する。アプリケーションは、メニューなどで印刷処理が選択されたとき、オペレーティングシステムに対して、作成された描画データを渡して印刷処理を依頼する。アプリケーションより印刷処理の依頼があったとき、オペレーティングシステムはアプリケーションより渡された描画データを、ホストコンピュータに接続されたプリンタが実行可能な印刷コマンドへ変換を行なうため、そのプリンタに対応したデバイスドライバの一例であるプリンタドライバを実行する。プリンタドライバは、オペレーティングシステムによって規定された手続きにより、描画データをオペレーティングシステムより受け取り、それを印刷コマンドへ変換する。そしてプリンタドライバはオペレーティングシステムの通信機能を使用して接続されたプリンタへ印刷コマンドの出力を行なう。

20

【0003】

プリンタは、機種ごとに、サポートしている用紙サイズ、インクの種類、用紙への記録方法、搬送可能な用紙の種類、用紙の搬送方法などの機能に違いがあり、それにより印刷制御方法、画像処理方法、使用できるプリンタコマンドのパラメータが異なる。そのため、アプリケーションの描画データをプリンタの制御コマンドへ変換するプリンタドライバは、プリンタの機種別の機能特性を十分考慮したうえで設計、作成を行わなければならない。よって、プリンタの機能特性の違いをプリンタドライバの制御アルゴリズムに反映させる必要あり、プリンタがサポートしているコマンド体系が同じであったとしても、制御アルゴリズムを実行するプログラムの違いからプリンタドライバの実行モジュールを各プリンタ個別に作成して供給することが行われている。

30

【0004】

一方、プリンタの機能特性をデータベース化して、機種別に対応しなければならない処理の部分をプリンタドライバはそのデータベースを参照して得た情報に応じて内部アルゴリズムを切り替えることというプログラム方法がある。この場合、一つのプリンタドライバの実行モジュールで複数種類のプリンタをサポートすることが可能である。このようにして、機能特性の異なる複数種類のプリンタに対し、ひとつのプリンタドライバを製品として供給することも行なわれている。

40

【0005】

上述したように、プリンタとそのプリンタの制御を行うプリンタドライバは対応関係があり、使用しているプリンタに適したプリンタドライバをホストコンピュータにインストールする必要がある。プリンタドライバのインストールやアンインストールの作業は、イン

50

ストーラと呼ばれるプログラムによって行っている。

【0006】

【発明が解決しようとする課題】

すでにプリンタドライバがインストールされているホストコンピュータにプリンタドライバをインストールするとき、プリンタドライバのファイル名に変更がない場合は、ファイルの重ね書きを行えばよく、問題は生じない。しかしながら、例えば、プリンタドライバが一つのプリンタのみをサポートするのか、複数のプリンタ機種をサポートするのかについてプリンタドライバの仕様の変更があった場合など、プリンタドライバのファイル名を変更して対応しなければならない場合がある。プリンタドライバのファイル名に変更がある場合は、以前よりホストコンピュータにあったプリンタドライバを削除する必要がある

10

【0007】

この際、ドライバをアンインストール（削除）するための機能は、プリンタドライバのインストーラの一機能として提供するか、或いは、専用のアンインストールプログラムとして提供することが考えられ、ユーザは、それを使用して手動で煩雑なアンインストール作業を行なうことが必要となる。

【0008】

しかしながら、例えば、一機種をサポートしているタイプのプリンタドライバから、複数機種をサポートするタイプのプリンタドライバへ変更することが必要なシステムの移行作業が行われる場合がある。この場合には、複数機種のプリンタをホストコンピュータに接続して使用しているときは、現在接続されているプリンタが新しい複数機種サポート型のプリンタドライバによりサポートされているかどうかを正確にユーザが一々把握した上で、既存の不要な機種用のプリンタドライバを選択的に削除するという作業を手動で行わなければならないであった。あるいは、一度すべてのプリンタドライバを削除してから、プリンタドライバをインストール（導入）し直すなどの面倒な作業が必要であった。

20

【0009】

本発明は上記課題を解決するためになされたものであり、不要となったプリンタドライバを自動的に認識して削除することにより、プリンタドライバのインストール作業を簡便化することができるインストーラの提供を目的とする。

【0010】

30

【課題を解決するための手段】

上記目的の少なくとも一つを達成するため、本発明は、例えば以下の構成を備える。

【0011】

請求項1に記載の発明は、周辺装置に接続可能な情報処理装置にデバイスドライバを管理する管理方法であって、情報処理装置内のデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報を取得する取得ステップと、前記取得手段が取得したデバイスドライバのバージョン情報とデバイスドライバが制御可能な周辺装置の情報から、特定の周辺装置に対応し、複数のバージョンが存在し得るデバイスドライバのうち、情報処理装置の記憶装置に存在するデバイスドライバのうち、不要となったデバイスドライバを認識する認識ステップと、前記認識手段によって不要と認識されたデバイスドライバの削除を制御する削除制御ステップを有することを特徴とする。

40

【0012】

請求項2に記載の発明は、前記認識ステップは、周辺装置を特定する特定手順と、前記特定手順により特定された周辺装置を制御可能なデバイスドライバのうち、最もバージョンが新しいデバイスドライバを不要ではないと判断する判断手順とを備え、前記特定手順により特定された周辺装置に対応するデバイスドライバに対して繰り返し前記判断手順における処理を行うことを特徴とする。

【0013】

請求項3に記載の発明は、前記認識ステップでは、周辺装置とデバイスドライバの情報を含むテーブルデータとして管理し、該テーブルデータより、不要となったデバイスドライ

50

バを求めることを特徴とする。

【0014】

請求項4に記載の発明は、新規にデバイスドライバを情報処理装置に追加した後に、前記取得ステップと、前記認識ステップと、前記削除ステップと、実行することを特徴とする。

【0015】

請求項5乃至請求項9に記載の発明は、請求項1乃至4に記載の発明の装置又はプログラムの発明であるので省略する。

【0016】

【発明の実施の形態】

以下、図面を参照して本発明の実施形態を詳細に説明する。

【0017】

図1はプリンタドライバのインストール先であるコンピュータとプリンタとからなるプリントシステムを示すブロック図である。図1において、1はホストコンピュータであり、イーサネット(R)やUSBなどの外部バス10(通信バス)を介してプリンタA11、プリンタB12が接続されている。図1ではプリンタは2つのみ図示したが、通信バス10を介して任意の数のプリンタが接続可能である。

【0018】

プリンタA11およびプリンタB12は、それぞれプリンタコントローラ13およびプリンタエンジン14を備えるインクジェットプリンタであり、種々のサイズのプリント媒体に画像、文字、グラフィクス等を実行して出力することができる。ここで、プリンタコントローラ13はプリンタ全体の制御を実行する制御回路およびプログラムを意味し、一方、プリンタエンジン14はインクジェットプリンタヘッドの走査やプリント媒体の搬送を行う機構そのものを意味するものである。プリンタ11およびプリンタ12は、それぞれインクジェットカートリッジを備え、インクジェットカートリッジはインクジェットヘッドのヘッドに供給するインクを貯留したインクタンクを色毎に組合せて一体化したものである。

【0019】

ホストコンピュータ1において、2はCPUであり、起動時ROM3に格納されたプログラムによりハードディスクやCD-ROMなどの外部記憶装置6に格納されたオペレーティングシステム(以下、図面ではOSとすることがある。)をRAM4にロードし実行してシステムを制御する。5はCRTあるいはLCDなどの表示装置であり、CPU1の制御にしたがって各種の表示を行うことができる。7はキーボード、マウスなどの入力装置であり、CPU1への各種入力を行う。9はホストコンピュータ内部バスを表し、CPU2、ROM3、RAM4、表示装置5、外部記憶装置6、入力装置7、外部I/F8との間でデータのやり取りを可能にするものである。

【0020】

外部記憶装置6には、オペレーティングシステムの他にアプリケーションプログラムやプリンタドライバなどの各種プログラムが格納されており、CPU1は種々のアプリケーションプログラムを実行し、オペレーティングシステムの制御に従い、プリンタA11、プリンタB12に対応したプリンタドライバを実行して、アプリケーションプログラムの描画データをプリンタA11、プリンタB12が実行可能なプリンタ制御コマンドに変換する。プリンタドライバの制御にもとづきCPU1が生成したプリンタ制御コマンドは、I/F8により、通信バス10に出力され、プリンタA11またはプリンタB12へ送信される。プリンタA11またはプリンタB12はこのようにしてホストコンピュータ1から送信されたプリンタ制御コマンドの受信を行い、印刷を行う。

【0021】

図2は、コンピュータのオペレーティングシステムにおける印刷処理に関するプログラム実行モジュールの内部ブロック図を表す。アプリケーション15は、あらかじめユーザの操作により作成された描画データを、オペレーティングシステム16に渡すとともにオ

10

20

30

40

50

オペレーティングシステム 16 に対し印刷処理の依頼を行う。オペレーティングシステム 16 はアプリケーションから渡された描画データを印刷スプールファイル 17 にセーブする。アプリケーションから描画データをすべて受け取り印刷スプールファイル 17 に保存した後、オペレーティングシステム 16 は出力先のプリンタに対応したプリンタドライバ 18 をロードする。オペレーティングシステム 16 は印刷スプールファイル 17 から描画データをロードし、プリンタドライバ 18 に渡し処理を依頼する。プリンタドライバ 18 は渡された描画データをプリンタコマンドへ変換して、オペレーティングシステム 16 の通信機能を利用してプリンタ 19 へ出力する。プリンタ 19 ではこのようにして生成された印刷コマンドにもとづいて印刷を行う。

【0022】

図 8 は、本実施形態のインストーラが実行されるときの実行環境例のブロック図を示す。図 8 はプリンタドライバ 1、プリンタドライバ 2 がすでにホストコンピュータ上にインストールされていた状態で、新たにプリンタドライバ 3 がインストールされた状態を示している。インストーラ 20 は、インストール作業の実行を担当するインストールメインモジュール 21 と、プリンタドライバのバージョンとサポートプリンタの情報を取得して処理を行うバージョン管理モジュール 22 で構成される。プリンタドライバ 1 は、バージョンが 1. 0. 0 の制御プログラムによるプリンタドライバであり、プリンタ A をサポートする。プリンタドライバ 2 は、バージョン 1. 1. 0 の制御プログラムを持つプリンタドライバであり、プリンタ D をサポートする。プリンタドライバ 3 は、バージョン 2. 0. 0 の制御プログラムを持つプリンタドライバであり、サポート情報テーブルにセーブされたサポート機種情報のプリンタをすべてサポートすることが可能なプリンタドライバである。

【0023】

図 5 は、サポート情報テーブルのデータ構造を表す。サポート情報テーブルは、プリンタドライバがサポートしているプリンタの機種数と機種の情報を持つ。図 5 の表のプリンタモジュール 3 の欄が、プリンタドライバ 3 に関するものであり、サポート可能なプリンタのデータが保存されている。本実施例では、プリンタドライバ 3 は、プリンタ A、プリンタ B、プリンタ C の 3 機種を制御可能である。

【0024】

本実施形態のインストーラが実行されると、まず、インストールメインモジュール 21 がホストコンピュータの CPU により実行される。図 6 は、インストールメインモジュール 21 の処理を表すフローチャートである。ステップ S1 において、プリンタドライバをオペレーティングシステムで規定された所定の場所にインストールする。ステップ S2 において、バージョン管理モジュール 22 を呼び出し、各プリンタドライバのバージョンとサポートしているプリンタ機種情報を取得して、プリンタドライバ管理テーブル B を作成させ取得する。

【0025】

図 7 は、ステップ S2 の詳細であり、バージョン管理モジュール 22 の処理を表すフローチャートである。ステップ S9 において、ホストコンピュータの外部記憶装置上に存在するプリンタドライバの検索を行なう。この処理は、例えば、プリンタドライバをオペレーティングシステムに規定された場所の範囲内で、あらかじめ決めておいたディレクトリの中にインストールしておくものとし、バージョン管理モジュール 22 がそのディレクトリの中の実行ファイルを検索することにより行われる。ステップ S10 において、検索で見つけたプリンタドライバの総数を変数 n に格納する。ステップ S11 で、カウンタ用の変数 i を 0 に初期化する。ステップ S12 で、カウンタ変数 i とプリンタドライバの総数 n を比較する。 $i < n$ のときは、ステップ S13 に進む。ステップ S13 において、プリンタドライバのバージョン情報を取得する。例えば、Mac OS X (商標) のような OS においては、プリンタドライバはバンドルと呼ばれるディレクトリ階層構造で構成される。ディレクトリ下の所定の場所に所定のファイル名のファイルがあり、その中にバージョン番号が記述されている。あるいは、Mac OS 9 (商標) のようなオペレー

10

20

30

40

50

ィングシステムでは、リソースと呼ばれるファイル領域にバージョン情報が記述されている。ステップS13では、例えば、アプリケーションとして提供されるインストーラが、これらOSが管理する情報を参照してバージョン情報を取得する。

【0026】

ステップS14では、インストーラは、i番目のプリンタドライバのサポート情報テーブルを検索する。例えば、Mac OS X (Apple社の商標) 用ドライバのようなディレクトリ構造で管理されるプリンタドライバは、プリンタドライバのディレクトリ階層下の所定の場所にサポート情報ファイルを格納されるものとする。別のオペレーティングシステムのプリンタドライバでは、リソースと呼ばれるファイル領域にサポート情報を保存しておくこともできる。ステップS14では、そのようなサポート情報テーブルの検索処理を行う。サポート情報テーブルが見つかった場合、ステップS15へ進む。ステップS15において、サポート情報テーブルの中に記述されたそのプリンタドライバが制御可能なプリンタの機種を取得する。S14において、サポート情報テーブルが見つからなかった場合、S16に進む。S16では、他の手段によりプリンタドライバがサポートするプリンタ機種情報を取得する。例えば、プリンタドライバのファイル名にプリンタの機種名が含まれるようなときは、バージョン管理モジュールはファイル名からプリンタドライバがサポートする機種名称を取得することができる。また、ファイル名にプリンタの機種名称が含まれていない場合でも、バージョン管理モジュールが、プリンタドライバのファイル名とプリンタ機種を対応付けるテーブルを持ち、これを参照することによってプリンタドライバが制御可能なプリンタの機種を知ることができる。

【0027】

ステップS17において、i番目のプリンタドライバがサポートするプリンタ機種情報とプリンタドライバのバージョン番号により、モジュール管理テーブルAを更新する。

【0028】

図9、図10はモジュール管理テーブルAの構造を表す図である。モジュール管理テーブルAは、プリンタの機種を表す情報、プリンタドライバのバージョン番号、プリンタドライバを識別するための情報を持つ。プリンタドライバを識別するための情報はオペレーティングシステムで管理されるプリンタドライバの参照情報である。

【0029】

図8はステップS17の詳細を表すフローチャートである。ステップS20において、現在処理対象となっているプリンタドライバがサポートするプリンタの機種数を変数mに代入する。ステップ21において、カウンタ変数jを0に初期化する。ステップ22において、変数mと変数jの大小関係を比較する。j<mのときは、ステップS23へ進み、プリンタドライバがサポートするプリンタのうち、j番目のプリンタに関してステップS23以降の処理を行う。ステップS23において、j番目のプリンタがプリンタドライバ管理テーブルAに登録済であるかどうか調べる。ステップS23で、j番目のプリンタがプリンタドライバ管理テーブルAに存在しないときは、そのプリンタの情報をモジュール管理テーブルAに登録するため、ステップS25へ進む。j番目のプリンタがプリンタドライバ管理テーブルAに存在するときは、ステップS24へ進む。ステップS24において、対象となっているj番目のプリンタに関して、プリンタドライバ管理テーブルAにすでに登録されているプリンタドライバのバージョンと、現在処理対象となっているプリンタドライバのバージョンを比較する。現在処理中のプリンタドライバのバージョンの方が大きいときは、ステップS25に進む。ステップS25では、プリンタドライバ管理テーブルAの該当プリンタの情報を現在処理対象となっているプリンタドライバの識別情報とバージョン情報で更新する。プリンタドライバ管理テーブルAにすでに登録されているプリンタドライバのバージョンよりも現在処理中のプリンタドライバのバージョンの方が小さいときは、S25のプリンタドライバ管理テーブルAの更新処理は行わない。ステップS26で、カウンタ変数jをインクリメントしてステップS22に戻り、処理対象となっているプリンタドライバの次のサポートプリンタに対して同様の処理を行う。ステップS22において、j<mでなくなり、すべてのプリンタについての処理が終了したときは、図8

10

20

30

40

50

のフローチャートの処理を終了して、図7のフローチャートのステップS18へ戻る。

【0030】

図7のフローチャートのステップS18において、カウンタ変数 i をインクリメントしてステップS12へ戻り、次のプリンタドライバに関して同様の処理を行う。ステップS12にて、 $i < n$ でなくなり、すべてのプリンタドライバについて処理が終了したときは、ステップS19へ進む。ステップS19において、プリンタドライバ管理テーブルAからプリンタドライバ管理テーブルBを作成する。

【0031】

図11、図12はプリンタドライバ管理テーブルBの構造を表す。プリンタドライバ管理テーブルBは、プリンタドライバ識別情報、プリンタドライバのバージョン番号、プリンタの機種数、プリンタの情報をもち、プリンタドライバをキーとしてプリンタドライバ管理テーブルAを並びかえたものである。そして、このテーブルを参照すれば、各バージョンのプリンタドライバを必要とするプリンタの機種がわかる。

10

【0032】

図8のプリンタドライバの構成において、バージョン管理モジュール22が図7、図8のフローチャートの処理を行い作成されたプリンタドライバ管理テーブルAとプリンタドライバ管理テーブルBが、図9、図11である。図9のプリンタドライバ管理テーブルAによれば、プリンタAをサポートする最もバージョンの大きいプリンタドライバはプリンタドライバ8であり、プリンタBをサポートする最もバージョンの大きいプリンタドライバはプリンタドライバ8であり、プリンタCをサポートする最もバージョンの大きいプリンタドライバはプリンタドライバ8であり、プリンタDをサポートする最もバージョンの大きいプリンタドライバはプリンタドライバ2であることがわかる。さらにこのテーブルをもとに作成された図11のプリンタドライバ管理テーブルBによれば、プリンタドライバ1を必要とするプリンタはなく、プリンタドライバ2はプリンタDのために必要であり、プリンタドライバ8はプリンタA、プリンタB、プリンタCのために必要であることがわかる。

20

【0033】

バージョン管理モジュールがプリンタドライバ管理テーブルBを作成後、インストールメインモジュール21の処理の図6のステップS3に戻る。ステップS3において、プリンタドライバ管理テーブルBに登録されたプリンタドライバの総数を変数 n に代入する。ステップS4において、カウンタ変数 i を0に初期化する。ステップS5において、変数 n と変数 i の大小関係を比較する。 $i < n$ のときは、ステップS6へ進む。ステップS6において、プリンタドライバ管理テーブルBの i 番目のプリンタドライバを必要とするプリンタの機種数を取得して機種数が0かどうかを調べる。機種数が0のときは、ステップS7へ進み、そのプリンタドライバの削除処理を行う。ステップS6において、 i 番目のプリンタドライバを必要とするプリンタの機種数が0でないときは、そのプリンタドライバの削除は行わない。

30

【0034】

ステップS8において、カウンタ変数 i をインクリメントして、ステップS5へ戻る。プリンタドライバ管理テーブルBに登録されているすべてのプリンタドライバの処理が終了して、ステップS5において、 $i < n$ ではなくなったときは、インストールメインモジュール21の処理を終了する。

40

【0035】

図8のようなプリンタドライバの構成では、プリンタドライバ管理モジュールBは図11のようであるので、プリンタ機種数が0であるプリンタドライバ1は、インストールメインモジュール21の実行後、削除される。

【0036】

図4は、本実施形態のインストーラが実行されるときの、他の実行環境例のフロー図を示す。図4は図8の環境にプリンタドライバ8がインストールされプリンタドライバ1が削除された状態において、さらに、プリンタドライバ4がインストールされたときの状態

50

を表している。プリンタドライバ4は、サポート情報テーブルを持ち、図5のプリンタドライバ4の欄に示すように、プリンタA、プリンタB、プリンタC、プリンタDの4機種を制御可能である。また、プリンタドライバ4の制御プログラムのバージョンは3.0.0である。

【0037】

このとき、図7、図8のフローチャートのバージョン管理モジュールが実行されることにより作成されるモジュール管理テーブルAは図10である。プリンタA、プリンタB、プリンタC、プリンタDをサポートする最もバージョンの大きいプリンタドライバは、ともにプリンタドライバ4であることがわかる。図10のプリンタドライバ管理テーブルAをもとに作成されるプリンタドライバ管理テーブルBを図12に示す。プリンタドライバ2、プリンタドライバ3を必要とするプリンタは存在せず、プリンタドライバ4を必要とするプリンタは、プリンタA、プリンタB、プリンタC、プリンタDであることがわかる。すなわち、図6のフローチャートにもとづくインストーラメインモジュールを実行すると、プリンタドライバ4がインストールされた後に、プリンタドライバ2とプリンタドライバ3は削除される。

10

【0038】

(他の実施形態)

本実施形態における各図に示す処理が、外部からインストールされるプログラムによって、各情報処理装置により遂行される。そして、その場合、CD-ROMやフラッシュメモリやFD等の記憶媒体により、あるいはネットワークを介して外部の記憶媒体から、プログラムを含む情報群をホストコンピュータに供給される場合でも本発明は適用されるものである。

20

【0039】

以上のように、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、又は、外部サーバ(図示省略)からダウンロードすることによって、そのシステムあるいは装置のコンピュータ(またはCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、本発明の目的が達成されることは言うまでもない。

【0040】

この場合、記憶媒体から読み出されたプログラムコード自体が本発明の新規な機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。プログラムコードを供給するための記憶媒体としては、たとえば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、DVD、CD-ROM、磁気テープ、不揮発性のメモ리카ード、ROM、EEPROM等を用いることができる。

30

【0041】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS(オペレーティングシステム)等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPU等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

40

【0042】

以上説明したように、本発明の上述の各実施形態によれば、プリンタドライバのバージョンや制御可能なプリンタなどの違いによりファイル名が異なるプリンタドライバがインストールされているホストコンピュータの環境において、プリンタドライバをインストールした後、不要となったバージョンの古いプリンタドライバを認識して削除することができるプリンタドライバのインストーラを提供できる。これにより、従来ユーザが手動で行っ

50

ていた不要となった古いバージョンのプリンタドライバを削除する作業をインストーラが行えるようになり、ユーザによるプリンタドライバのアンインストール作業を簡便化することができる。

【0048】

【発明の効果】

不要となったプリンタドライバを自動的に認識し、認識したドライバを削除するので、プリンタドライバのアンインストール作業を簡便化することができる。

【図面の簡単な説明】

【図1】コンピュータとプリンタとからなる印刷システムの構成の一例を示すブロック図である。

10

【図2】印刷システムにおいてオペレーティングシステムを介した印刷処理を表すブロック図である。

【図3】インストーラが実行される環境の例を表すブロック図である。

【図4】インストーラが実行される環境の他の例を表すブロック図である。

【図5】プリンタドライバが制御可能なプリンタの情報を持つサポート情報テーブルを示す図である。

【図6】インストーラメインモジュールの処理を表すフローチャートである。

【図7】バージョン管理モジュールの処理を表すフローチャートである。

【図8】図7において、プリンタドライバ管理テーブルAの更新処理の詳細を表すフローチャートである。

20

【図9】図8のインストーラ実行環境例において作成されるプリンタドライバ管理テーブルAを示す図である。

【図10】図4のインストーラ実行環境例において作成されるプリンタドライバ管理テーブルAを示す図である。

【図11】図8のインストーラ実行環境例において作成されるプリンタドライバ管理テーブルBを示す図である。

【図12】図4のインストーラ実行環境例において作成されるプリンタドライバ管理テーブルBを示す図である。

【符号の説明】

- 1 ホストコンピュータ
- 2 CPU
- 3 ROM
- 4 RAM
- 5 表示装置
- 6 外部記憶装置
- 7 入力装置
- 8 外部I/F
- 9 内部バス
- 10 外部バス
- 11 プリンタA
- 12 プリンタB
- 13 プリンタコントローラ
- 14 プリンタエンジン
- 15 アプリケーション
- 16 オペレーティングシステム
- 17 印刷スプールファイル
- 18 プリンタドライバ
- 19 プリンタ
- 20 インストーラ
- 21 インストールメインモジュール

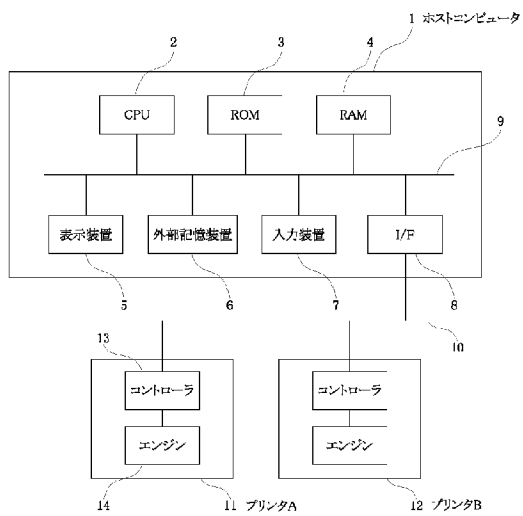
30

40

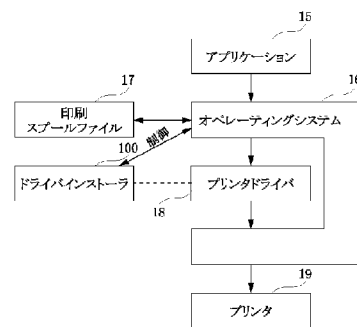
50

- 2 2 バージョン管理モジュール
- 2 3 プリンタドライバ 1
- 2 4 プリンタドライバ 2
- 2 5 プリンタドライバ 3
- 2 6 プリンタドライバ 4

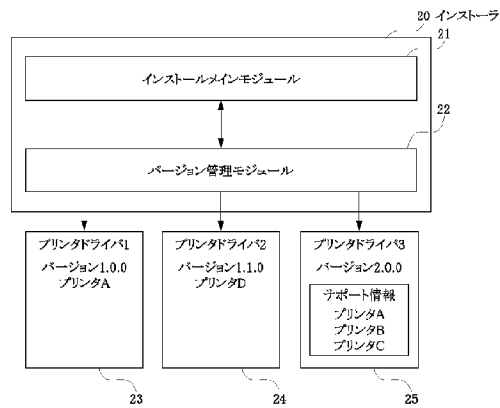
【図 1】



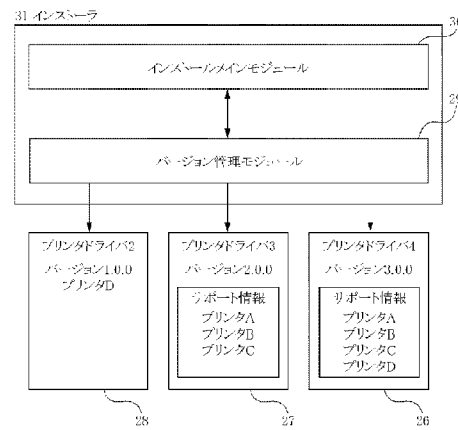
【図 2】



【図 3】



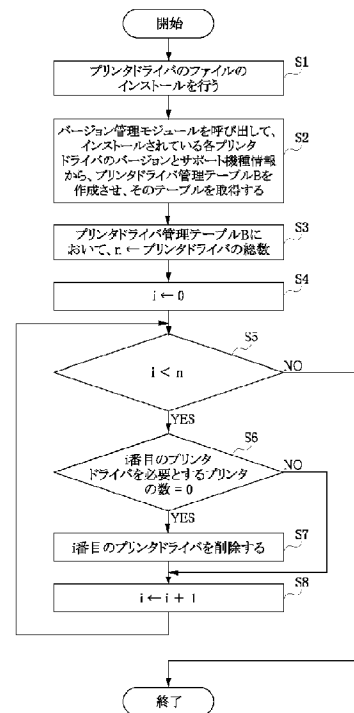
【図 4】



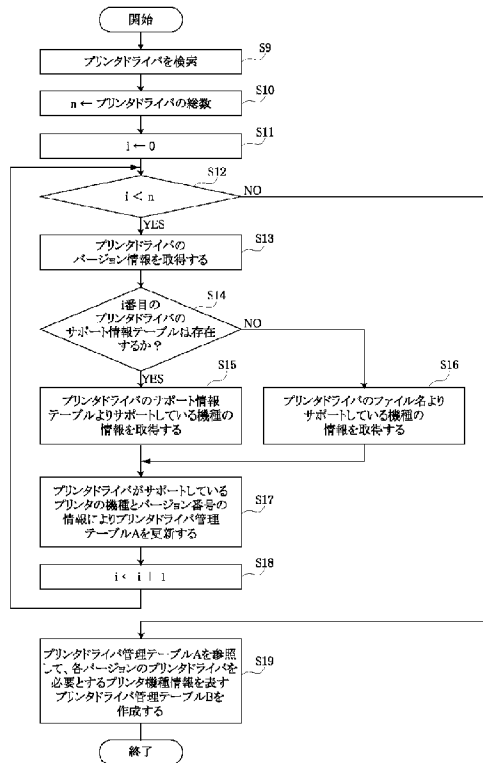
【図 5】

プリンタドライバ	バージョン番号	サポート情報	
		サポート機種数	サポートプリンタ
プリンタドライバ 3	2.0.0	3	プリンタ A プリンタ B プリンタ C
プリンタドライバ 4	3.0.0	4	プリンタ A プリンタ B プリンタ C プリンタ D

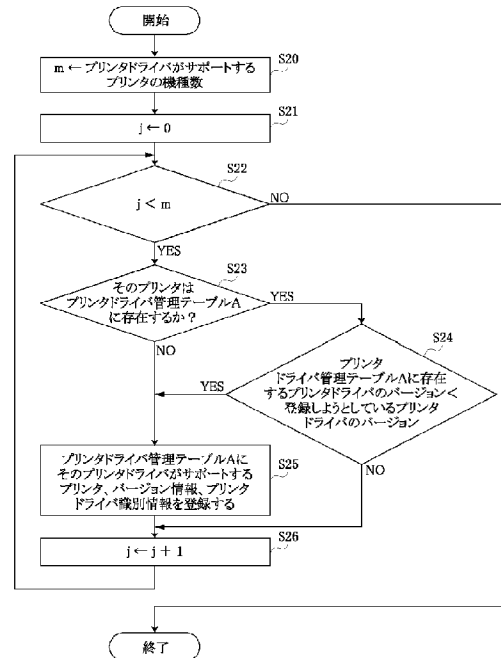
【図 6】



【図 7】



【図 8】



【図 9】

プリンタ	バージョン情報	プリンタドライバ識別情報
プリンタA	2.0.0	プリンタドライバ3
プリンタB	2.0.0	プリンタドライバ3
プリンタC	2.0.0	プリンタドライバ3
プリンタD	1.1.0	プリンタドライバ2

【図 10】

プリンタ	バージョン情報	プリンタドライバ識別情報
プリンタA	3.0.0	プリンタドライバ4
プリンタB	3.0.0	プリンタドライバ4
プリンタC	3.0.0	プリンタドライバ4
プリンタD	3.0.0	プリンタドライバ4

【図 1 1】

プリンタドライバ識別情報	バージョン番号	プリンタ機種数	プリンタ
プリンタドライバ1	1.0.0	0	なし
プリンタドライバ2	1.1.0	1	プリンタD
プリンタドライバ3	2.0.0	3	プリンタA プリンタB プリンタC

【図 1 2】

プリンタドライバ識別情報	バージョン番号	プリンタ機種数	プリンタ
プリンタドライバ2	1.1.0	0	なし
プリンタドライバ3	2.0.0	0	なし
プリンタドライバ4	3.0.0	4	プリンタA プリンタB プリンタC プリンタD